

(c)1986 PFP

MONITOR XL

Ein Maschinensprachmonitor für

**Atari-Computer
600XL(64K)/800XL/130XE**

MONITOR XL

Mit MONITOR XL haben Sie ein universelles Werkzeug zur Fehlersuche in Maschinenprogrammen erworben. Mehr noch, MONITOR XL kann zusammen mit BASIC verwendet werden, ohne daß der Speicher für BASIC-Programme verkleinert wird.

Somit ist MONITOR XL ideal zur Fehlersuche in BASIC-Programmen geeignet, die mit Maschinenunterroutinen arbeiten.

Laden des MONITOR XL

Schalten Sie die Stromversorgung des Disk-Laufwerkes und evtl. von Drucker und Interface ein. Nun die MONITOR XL Diskette in Laufwerk 1 stecken. Stromversorgung des Computers einschalten und warten, bis die Diskette eingelesen ist. Sie sehen die Copyright-Meldung auf dem Schirm, darunter meldet sich BASIC mit 'Ready'. Wird kein Basic benötigt, dann vor dem Einschalten des Computers die 'OPTION'-Taste drücken, bis die blaue Bildschirmfarbe sichtbar wird. In diesem Fall befinden Sie sich nach dem Boot-Vorgang direkt im MONITOR XL.

Aufruf von MONITOR XL

Haben Sie mit BASIC gebootet, dann können Sie den MONITOR XL jederzeit mit dem Befehl 'BYE' ('B.' genügt als Abkürzung) erreichen. Selbstverständlich können Sie diesen Befehl auch innerhalb eines Programms verwenden, das ist hervorragend, wenn man einige Speicherzellen an einer bestimmten Stelle des Programms (z.B. nach dem Ablauf einer USR-Routine) prüfen möchte.

RESET-Taste

MONITOR XL ist selbstverständlich gegen die RESET-Taste geschützt, d.h. RESET bringt Sie stets in den Monitor zurück.

BREAK-Taste und Joker

Bei einigen Befehlen, die eine Anfangs- und Endadresse benötigen, kann die Endadresse durch ein '-' Zeichen ersetzt werden. Dieses Zeichen (im folgenden JOKER genannt) wird von MONITOR XL als Adresse \$FFFF interpretiert, so daß als Endadresse die höchste Speicheradresse genommen wird. Mit Break kann eine Ausgabe des MONITOR XL jederzeit abgebrochen werden.

Zahlendarstellung

MONITOR XL kennt drei verschiedene Zahlensysteme: dezimal , hexadezimal (sedezimal) und binär. Hexadezimale Zahlen werden mit einem '\$'-Symbol gekennzeichnet , dezimale mit '#' und binäre Zahlen werden mit '%' bezeichnet. Wenn Sie eine Zahl als Teil eines Befehls eingeben , dann sollten Sie sich immer darüber im klaren sein, in welchem Zahlensystem die Eingabe erfolgt. Verwenden Sie daher immer das entsprechende Symbol am Anfang einer Zahl.

WICHTIG: Wenn Sie kein Symbol eingeben, dann wird die Eingabe als hexadezimale Zahl interpretiert. Bei dezimalen Zahlen immer '#' eingeben! Mit dem 'U'-Befehl können Sie Zahlen innerhalb aller drei Zahlensysteme umrechnen lassen.

Beispiele; #100 (=100), \$100 (=256), %100 (=4)

Befehlseingabe und Hilfe

Alle Befehle von MONITOR XL werden mit einem Buchstaben aufgerufen. Wenn zusätzliche Eingaben nötig werden, müssen diese durch ein Leerzeichen getrennt werden. Lesen Sie aber dazu die Beschreibungen der einzelnen Befehle, dort wird das Eingabeformat der einzelnen Befehle erläutert. Eine Liste aller Befehle von MONITOR XL können sie jederzeit durch die Eingabe von '?' (und <RETURN> drücken!) erhalten.

Mit '? P' können Sie diese Liste auch auf dem Drucker ausgeben. Wenn Ihnen ein Eingabefehler unterläuft, wird Ihnen das im Klartext mitgeteilt. Sie können alle gewohnten Editier-Tasten benutzen, um den Fehler zu korrigieren.

Beschreibung der einzelnen Monitorbefehle

Zuerst ist es dringend nötig, Klarheit über die verwendete Notation zu schaffen. Wird ein Parameter in der Syntax Beschreibung in eckigen Klammern ('[]') angegeben, so kann diese auch weggelassen werden. Ein Begriff in spitzen Klammern ('()') ist stets symbolisch zu verstehen, d.h. mit <Return> ist die RETURN-Taste gemeint und <Anfadr> steht stets für eine Adressangabe (mit einer Anfangsadresse, wie der Name vermuten läßt).

Hier eine kurze Zusammenstellung der wichtigsten Angaben:

<Anfadr>	: Adresse, in der ein Speicherbereich beginnt
<Endadr>	: Adresse, in der ein Speicherbereich endet
<Zieladr>	: Zieladresse z.B. für MOVE-Befehl
<Wert>	: Byte-Wert 0 bis \$FF (0-#255)
<Sektor>	: gültige Sektornummer
<String>	: Zeichenkette, stets in Anführungszeichen
<Filename>	: Name einer Datei im Atari-Format, von Anführungszeichen begrenzt ("D: FILENAME.EXT")
<A-Befehl>	: steht hier für einen Assemblerbefehl in mnemonischer Form (z.B. STA \$2C4)

Sonstige Symbole sind so gewählt, daß sie sich selbst erklären. Unter der Abkürzung 'PC' ist der gespeicherte Programmzähler zu verstehen, den Sie jederzeit mit 'V' prüfen können. Doch nun zu den eigentlichen Befehlen:

A - Assembler: Mit diesem Zeilenassembler können Sie Maschinenprogramme im Speicher korrigieren und auch kurze Programme eintippen. Alle Adressen müssen absolut eingegeben werden, d. h. Labels sind nicht zulässig. Bei BRANCH-Befehlen können entweder das absolute Sprungziel oder bei Vorwärtssprüngen der Offset (mit vorangestelltem '+' Zeichen) angegeben werden.

SYNTAX : A <Adresse> <A-Befehl>

BEISPIEL : A \$600 LDA #\$00
A \$602 BNE +\$1F

A \$604 STA (\$D0),Y

Sobald Sie eine Zeile eingegeben haben, wird diese auf Richtigkeit geprüft und durch den Disassembler wieder ausgegeben. Als Hilfe zur Eingabe von mehreren Assemblerzeilen wird in der nächsten Zeile ein neues A-Kommando mit der nächsten freien Adresse ausgegeben, so daß Sie nur noch den Assemblercode ergänzen müssen. Die Eingabe kann jederzeit mit <BREAK> abgebrochen werden. Im Assembler sollten Sie nur Hex-Zahlen mit vorangestelltem '\$'-Zeichen verwenden, um Zweideutigkeiten (besonders beim LDA-Befehl) mit dem '#'-Zeichen zu vermeiden.

B - Blood: Entspricht der DOS-Funktion 'L' und erlaubt, ein binäres File von der Diskette zu laden. ein solches File kann zuvor mit dem 'K'-Befehl von DOS 2.5 oder mit dem 'S'-Befehl (Save) von MONITOR XL erzeugt worden sein. Als Parameter wird ein Filename im normalen ATARI-Format benötigt.

SYNTAX : B <Filename>

BEISPIEL : B „D1:TEST.OBJ“

C - Compare Memory: mit diesem Befehl können zwei Speicherblöcke miteinander verglichen werden. Die unterschiedlichen Speicherzellen werden am Bildschirm ausgegeben.

Wird die Drucker-Option (mit 'P' am Ende des Befehls) angewählt, dann werden diese Adressen auch ausgedruckt.

SYNTAX : C <Anfadr> <Endadr> <Zieladr> [P]

BEISPIEL : C \$5000 \$5800 \$8000 P

Dieses Beispiel würde den Speicherblock ab Adresse \$5800 bis \$5800 mit dem Speicher-Inhalt von \$8000 bis \$8800 vergleichen und die Abweichungen sowohl auf dem Bildschirm als auch auf dem Drucker protokollieren.

D - Dump Memory: Gibt die Inhalte eines Speicherbereiches in hexadezimaler Schreibweise und im ATASCII-Code aus. Wiederum kann die Ausgabe auch auf dem Drucker erfolgen, dazu muß nur die Drucker-Option mit 'P' gesetzt werden.

SYNTAX : D <Anfadr> <Endadr> [P]

BEISPIEL : D \$6000 \$6100 P

Im Beispiel würde der Speicherbereich von \$6000 bis \$6100 als sog. HEX-DUMP sowohl auf dem Bildschirm als auch auf dem Drucker ausgegeben.

E - Error Message: Diese Funktion ist als Lebenshilfe für BASIC-Programmierer gedacht. Sie können damit zu jeder definierten Fehlernummer einen kurzen Text abrufen, der Ihnen die Ursache des Fehlers erläutert. Natürlich war es aus Gründen des Speicherplatz nur möglich, recht knappe Meldungen unterzubringen, aber im Regelfall wird das zur Diagnose reichen. Wenn Ihnen BASIC einen Fehlercode liefert, dann aktivieren Sie mit 'BYE' den Monitor und geben 'E' gefolgt von der Nummer des Fehlers ein und voila – haben Sie einen Text dazu. Sie sollten allerdings nicht vergessen, vor die Zahl ein '#' (für dezimal) zu schreiben.

SYNTAX : E <Wert>

BEISPIEL : E #5

ergibt: Fehler -5: String Länge

F - Fülle Speicher: Ein angegebener Speicherbereich wird mit einem bestimmten Wert gefüllt.

SYNTAX : F <Anfadr> <Endadr> <Wert>

BEISPIEL : F #1536 #1761 \$FF

Das Beispiel schreibt den Wert \$FF in die PAGE 6.

G - Gehe nach Adresse: Ein Maschinenprogramm wird aufgerufen. Wird keine Adresse angegeben, so wird der unter 'V' (s. dort) gespeicherte Wert des PCs genommen. Alle CPU internen Register werden mit den in 'V' eingestellten Werten geladen.

SYNTAX : G [<Anfadr>]

BEISPIEL : G \$E477

H - Hunt: Sucht nach einer angegebenen Zeichenfolge in einem bestimmaren Speicherbereich. Als Suchobjekte können sowohl Zeichenketten als auch Hexwerte dienen, sogar String/Hex-Kombinationen sind möglich. Alle Adressen im angegebenen Bereich, deren Inhalt mit den gesuchten Werten übereinstimmen, werden auf dem Bildschirm und <fil& Option angewählt> auch auf dem Drucker ausgegeben.

SYNTAX : H <Anfadr> <Endadr> <String/Wert>...[P]

BEISPIEL : H \$700 \$2000 "FREE SECTORS"
H \$7000 \$7800 \$FF \$A9 \$00
H \$5000 \$8000 "Hugo" \$9B P

I - Disk Inhalt: Das Disketteninhaltsverzeichnis wird auf dem Bildschirm (auch Drucker möglich) ausgegeben; dabei werden zwei Einträge pro Zeile gelistet. Wird kein Parameter angegeben, so werden alle Einträge angezeigt, im anderen Fall werden (wie beim DOS) nur übereinstimmende Einträge gelistet.

SYNTAX : I [<Filename>] [P]

BEISPIEL : I <Inhalt Laufwerk 1>
I "D:*.OBJ" (alle Files mit .OBJ)
I "D2:*.*" (Inhalt Laufwerk 2)

L - Load: Lädt ein binäres File. Im Gegensatz zu Bload werden INIT und RUN-Adressen nicht ausgeführt (d.h. Das Programm wird nicht gestartet).

SYNTAX : L <Filename>

BEISPIEL : L "D:TEST.OBJ"

M - Move Memory: Ein bestimmter Speicherblock wird an eine andere Stelle kopiert.

SYNTAX : M <Anfadr> <Endadr> <Zieladr>

BEISPIEL : M \$2000 \$2800 \$5000

Kopiert den Speicherblock von \$2000 bis \$2800 (einschließlich) in den Bereich von \$5000 bis \$5800.

P - Put Sectors: Mit diesem Befehl kann ein Speicherblock direkt in einzelnen Sektoren einer Diskette abgelegt werden. Sie geben dazu die Anfangsadresse des Speicherblocks, den Startsektor und die Anzahl der Sektoren an. Die Länge des zu schreibenden Bereiches ergibt sich automatisch aus der Anzahl der

Sektoren (zu je 128 Bytes). VORSICHT: Wenn Sie mit dieser Funktion auf DOS-Disketten schreiben, besteht die Gefahr, daß einzelne Files oder auch (Horror of Horrors) das Inhaltsverzeichnis zerstört werden kann. Daher Vorsicht!

SYNTAX : P <Anfadr> <Sektor> <Anzahl>

BEISPIEL : P \$2000 #700 #4 (Vorsicht!!!)

Das Beispiel schreibt den Bereich von \$2000 bis \$21FF in die Sektoren 700 bis 703 (dez.).

Q - Quit: Rücksprung zum Ausgangsprogramm, beim Booten mit BASIC gelangen Sie ins BASIC, sonst ins DOS.

SYNTAX : Q

BEISPIEL : Q

R - Read Sectors: Quasi das Gegenstück zum 'P'-Befehl. Damit kann eine Anzahl von Sektoren in den Speicher eingelesen werden.

SYNTAX : R <Anfadr> <Sektor> <Anzahl>

BEISPIEL : R \$2000 #700 #4

Liest den Inhalt der Sektoren 700 bis 703 (dez.) in den Bereich von \$2000 bis \$21FF.

S - Save: Abspeichern eines Programms als Binärfile. Solche Files können mit 'B', 'L' oder im DOS mit 'L' wieder eingelesen werden. Zusätzlich kann jedes File mit einer INIT und RUN-Adresse versehen werden (näheres im DOS Handbuch).

Wird nach dem Filenamen ein 'A' (für APPEND) angegeben, dann wird das File an ein bereits bestehendes mit gleichem Namen 'hinten' angehängt.

SYNTAX : S <Filename> [A] <Anfadr> <Endadr> [,<Init>] [,<Run>]

BEISPIEL : S "D:TEST.OBJ" \$600 \$600,\$600 (mit INIT)
S "D2TEST.OBJ" A \$2000 \$3000,, \$2000 (APPEND m. RUN)

T - Trace: Das ist der wohl leistungsfähigste Befehl von MONITOR XL. Damit können Sie ein Maschinenprogramm Schritt für Schritt abarbeiten lassen und bekommen jeden ausgeführten Befehl samt Registerinhalte am Schirm angezeigt. Die Ausgabe kann mit <CONTROL>-1 angehalten und wieder aufgenommen werden, Abbruch ist mit <BREAK> jederzeit möglich. MONITOR XL beendet die TRACE-Funktion, wenn ein nicht definierter Opcode oder ein BRK-Befehl auftritt. Wird keine Startadresse angegeben, so wird der derzeitige PC (aus dem 'V'-Befehl) genommen. Auch hier kann die Druckeroption eingeschaltet werden.

SYNTAX : T [<Anfadr>] [P]

BEISPIEL : T \$600 P

U - Umwandlung: Erlaubt die Umrechnung von Zahlen zwischen den drei gängigen Zahlensystemen (Dez, Hex und Binär). Als Parameter kann ein Ausdruck angegeben werden, in dem Zahlen aller drei Systeme mit den Grundrechenarten (+, -, *, /) miteinander verknüpft werden. Das Ergebnis wird dann dezimal, hex und binär ausgegeben. Die Ausdrücke werden von links nach rechts ohne Beachtung von Prioritäten ausgewertet. Ideal ist diese Funktion auch als Hilfe für BASIC-Programmierer.

SYNTAX : U <Wert> <Operator> <Wert>... (<Operator>:= +,-,*,/)

BEISPIEL : U #356 / \$30 - %11001

V - View Register: Tritt während der Ausführung eines Maschinenprogramms ein BRK-Befehl (ein sog. Break-Point) auf, so geht die Kontrolle auf MONITOR XL zurück, und die Inhalte der CPU-internen Register werden gerettet. Diese Registerwerte können Sie sich mit dem V-Befehl anzeigen lassen und auch durch andere Werte ersetzen. Zur Eingabe von neuen Werten gehen Sie bitte wie folgt vor: Zuerst 'V' <Return> eingeben, dann die angezeigten Werte durch überschreiben Ändern und wieder <Return> drücken. Nach gleichem Muster können Sie auch beim Trace- und Einzelschrittmodus vorgehen. Wenn ein 'G'-Befehl ausgeführt wird, werden vor dem Aufruf des Maschinenprogrammes die geretteten Registerinhalte wieder in die Register zurückgeschrieben, so daß Sie nach einem BRK- Befehl einfach mit 'G' im Programm fortfahren können.

SYNTAX : V <Registerinhalte wie angezeigt>

BEISPIEL : V 00 00 00 (Akku, X und V-Reg. auf Null)

W - Einzelschritt (Walk): Entspricht dem T-Befehl, mit dem Unterschied, daß nur der nächste Befehl ausgeführt wird. Wird nur 'W' eingeben, wird der Inhalt des gespeicherten PCs (s. 'V') genommen.

SYNTAX : W [<Anfadr>] [P]

BEISPIEL : W #30000 P

X - Disassembler: Mit diesem Befehl können Programmteile in mnemonischer Schreibweise dargestellt werden. Wird nur eine Anfangsadresse eingegeben, so wird auch nur ein Befehl disassembliert. Als Besonderheit können bereits disassemblierte Zeilen durch überschreiben der Hexcodes und anschließendem Druck auf <Return> verändert werden. Die Mnemonics werden dann berichtigt.

SYNTAX : X <Anfadr> [<Endadr>] [<P>]

BEISPIEL : X \$700 \$780

Y – Druckerbefehl: Damit können Sie Texte und Steuerzeichen an den Drucker ausgeben. Sehr nützlich, wenn Sie eine Überschrift zu einem Hex-Dump brauchen, oder die Schriftgröße verändern wollen. Als Parameter sind Zeichenketten und Hex-Werte zulässig.

SYNTAX : Y <Zeichenkette od. Hex-Wert>...

BEISPIEL : Y #27 "M" "Elite-Schrift ein"

Z - Zum DOS: Ruft das DOS auf.

SYNTAX : Z

BEISPIEL : Z

: - Speichern: Dient zum direkten Verändern von Speicherzellen. Als erster Parameter nach ':' muß (ohne Leerzeichen dazwischen) die Adresse stehen, ab der die Daten abgelegt werden sollen. Dann folgen die einzelnen Werte, die in dezimaler und hexadezimaler Schreibweise eingegeben werden können. Zu Ihrer Erleichterung gleicht das Eingabeformat dieses Befehls dem Ausgabeformat von 'D' (Memory-Dump), so daß Sie mit 'D' erstellte Memory-Dumps direkt mit dem Cursor überschreiben und mit nachfolgendem <Return> übernehmen können.

SYNTAX : :<Anfadr> <Wert> <Wert>...

BEISPIEL : :8000 FF 03 #100

? - Hilfe: zeigt Ihnen eine kurz Übersicht aller verfügbaren Befehle von MONITOR XL an.

SYNTAX : ? [P]

BEISPIEL : ? P (druckt Befehlsliste aus)

BEISPIEL

Ein kleines Beispiel wird Ihnen helfen sich mit den wesentlichsten Befehlen von MONITOR XL vertraut zu machen. Geben Sie dazu folgendes mit dem Zeilenassembler ein (nach der letzten Eingabe <BREAK> drücken):

```
A $600 LDX #$00
A $602 STX $2C8
A $605 INX
A $606 INX
A $607 CPX #$0C
A $609 BNE $602
A $60B BRK
```

Mit 'D 600 610' können Sie sich nun das Produkt dieser Eingabe als Memory-Dump ansehen, oder per Disassembler mit x 600 610 in mnemonischer Form anzeigen lassen.

Versuchen Sie nun die Trace-Funktion mit 'T 600', Sie sehen, wie die Befehle nacheinander ablaufen und die Rahmenfarbe langsam heller wird. Einzelschritte können Sie mit 'W 600' erhalten, der nächste Befehl wird durch Eingabe von 'W' <Return> ausgeführt. In Echtzeit können Sie das Programm mit 'G 600' starten. Augenblicklich wird die Rahmenfarbe hell und die Registerinhalte nach Ablauf des Programms dank dem BRK-Befehl angezeigt.

MONITOR XL intern

MONITOR XL liegt im Bereich von \$A000 bis \$BC00 unter dem BASIC-ROM verborgen. Zusätzlich wird der Kassettenbuffer (\$400-\$480) durch einige kurze zum Umschalten des BASIC-ROMs nötige Programmteile belegt. Diesen Bereich bitte nicht anderweitig verwenden, auch sind Kassettenoperationen bei der Verwendung von MONITOR XL nicht erlaubt.

MONITOR XL braucht auch noch etwas Platz für seine internen Variablen. Dies wurde gelöst, indem ein Teil der Page 0 (\$82 bis \$D3) und Page 2 beim Aufruf des Monitors in interne Buffer gerettet werden, die erst beim Verlassen des Monitors zurückgeschrieben werden. Auf diese Weise kann MONITOR XL auch problemlos mit BASIC zusammenarbeiten.

MONITOR XL kopiert während des Bootvorganges das OS-ROM in den darunterliegenden Speicher und richtet den sog. 'BlackBoard'-Vektor auf seine Einsprungadresse (daher der Aufruf über BYE). Nebenbei: Da das Betriebssystem bei der Verwendung von MONITOR XL im RAM liegt, kann es auch verändert werden - mit der entsprechenden Vorsicht.

Zum Thema Raubkopien:

MONITOR XL ist ein Werkzeug für Programmierer, und ist daher in keinsten Weise kopiergeschützt, d.h. Sie können MONITOR XL auf Ihre Arbeitsdiskette kopieren, mit Ihrem Lieblings-DOS verwenden etc. Leider - und das muß nach bisheriger Erfahrung gesagt werden - verleitet diese offene und anwendungsfreundliche Art auch sehr zur nicht autorisierten Weitergabe des Programmes.

Daher: Wenn ein Bekannter von Ihnen auch MONITOR XL verwenden möchte, dann geben Sie ihm anstatt einer Kopie der Diskette doch den Hinweis, daß das Programm komplett mit Anleitung zu einem wirklich günstigen Preis angeboten wird, den sich jeder leisten kann (DM 19.80). Es wäre doch unfair, wenn Sie den Autor, der mit MONITOR XL wirklich sein Bestes gegeben hat, um seine Lizenzen bringen würden.

Für alle, die tiefer in die Geheimnisse der Maschinensprache eindringen wollen. Mit MONITOR XL haben Sie folgende Funktionen zur Verfügung:

- Assemblerprogramme können Schritt für Schritt ausgeführt werden. Nach jedem Befehl werden Registerinhalte und Prozessorflags angezeigt.
- Eingebauter Zeilenassembler und Disassembler zum Eingeben und Korrigieren von Maschinenprogrammen
- Laden und Speichern von Binär-Files
- Deutsche Fehlermeldungen auch für BASIC
- Eingebaute 'HILFE'-Funktion
- Braucht keinen BASIC-Speicherplatz
- Ausführliche Anleitung mit Beispielen

100%. Maschinensprache! Zum sagenhaft günstigen Preis! Für ATARI-Computer 600XL(64K)/800XL/130XE

PETER FINZEL PRODUCTIONS, FÜRTH/BAY.